



1076.41035PX1

## ACCESSING FUNCTIONALITIES IN HYPERMEDIA

### CROSS-REFERENCE TO RELATED APPLICATION

This application is a continuation-in-part of United States Patent Application  
5 Serial No. 10/024,717 filed on 21st December 2001 entitled "Accessing Functionalities  
in Hypermedia", which application is incorporated herein by reference in its entirety.

### Field of the Invention

This invention relates to accessing functionalities in hypermedia and has  
10 particular but not exclusive application to displaying links independently of their location  
within a hypertext document.

### ~~Background of the invention~~ Description of the Prior Art

Recommendations have been developed to provide a wireless application  
15 protocol (WAP) to promote common standards and specifications for data services that  
operate over wireless communication networks. For example, mobile devices such as  
telephone handsets can be WAP enabled to allow a user to access remote servers.  
Other mobile data communication services have also been developed such as i-mode.  
Data content is provided in a mark up language, similar to conventional hypertext mark  
20 up language (HTML), known as wireless mark up language (WML). This is configured to  
allow data to be displayed as a deck of individual cards which are of the size suited for  
display on the relatively small display screen usually available on a mobile device such

as a mobile telephone handset or personal digital assistant (PDA).

The mobile device runs a browser to display data from the server and the user controls navigation between different decks and cards with keys on the mobile device. Typically, the mobile device has a smaller number of keys than a conventional personal  
5 computer and does not have a mouse, and so navigating with the browser of the mobile device gives rise to a problem in terms of ergonomics, in relation to the ease of browser controllability for the user. For example, the deck of cards may include a number of different links to different network locations, each specified by individual universal resource locator (URL). The user may have difficulty in locating the URLs provided in a  
10 particular deck of cards because they may be scrolled away from the currently visible content and so, hitherto, the browser has been configured to provide a list of such links separately from the display of the individual cards so that the user can operate an menu mechanism such as an “options list” in Nokia phones to select the list of links. This has the advantage that the user does not need to browse through the entire deck of cards to  
15 determine the links available. In the past, the provision of the options list has been achieved using the “DO” element. This is a conventional feature of WML, which as explained in more detail hereinafter, causes the user interface to perform predetermined user controlled functions. For a full description of the DO element, reference is directed to “Official Wireless Application Protocol” Wireless Application Protocol Forum Limited,  
20 Wiley, 1999 (ISBN0-471-32755-7) pp 87-89. In order to provide the options list, links that are marked with a DO element are provided in the options list. Then, having selected the list, the user can select an individual link from the list. Thus, the user does

not need to browse through the entire deck of cards to identify the links contained within it.

Recently, it has been proposed to provide browsers compatible with XHTML rather than WML. XHTML is a development of HTML 4, and is described in XHTML 1.0:

5 The Extensible Hypertext Mark Up Language (second edition) — a reformulation of HTML 4 in XML 1.0, W3C working draft 4 October 2001 published by W3C. XHTML is suitable for use with alternative platforms to those conventionally used for HTML, such as fixed work stations and personal computers. XHTML differs from WML in that the content is not arranged in a deck of cards but rather in documents more akin to  
10 conventional HTML for use with conventional browsers on personal computers. Although the syntax rules and the specifications differ between XHTML and HTML, they contain the same elements and neither of them includes the DO element found in WML. Therefore, it is not possible to provide the previously described options list that can be achieved in WML, for a XHTML browser.

15 It is an object of the invention to overcome this problem.

## SUMMARY OF THE INVENTION

According to the invention there is provided a method of accessing functionalities in hypermedia to be parsed and rendered by a user agent, the hypermedia including at  
20 least one element that has a predetermined attribute whereby a dynamically assignable keyboard shortcut for the user agent actuates a predetermined functionality associated with the element, the method comprising: parsing the hypermedia, collating data

corresponding to those elements in the hypermedia that support said predetermined attribute, and rendering a display of the collated data.

5 The invention also includes a browser for hypermedia marked up with at least one element that has a predetermined attribute whereby the actuation by a user of a selected individual keyboard control, actuates a predetermined functionality associated with the element, the browser including software to provide: a parsing process to parse the hypermedia, a process to identify data corresponding to those elements in the hypermedia that support said predetermined attribute, and a process to render a display  
10 of the data.

The predetermined attribute may be an acceskey function. The browser may be configured for use with XHTML.

The browser may be provided in a mobile device such as a mobile telephone handset or a PDA.

15 The invention has the advantage of providing a list of elements independently of the hypermedia containing them, by the use of an attribute such as an acceskey.

The invention also includes a device for rendering hypermedia received from a remote server, the device including a processor for the hypermedia and a user interface including a display device and a keyboard with a plurality of keys operable in a first  
20 mode to enter associated alphanumeric data, and operable in a second mode to actuate respective keyboard shortcuts dynamically assigned thereto by elements in the hypermedia, the processor and the display device being operable in a first display

configuration to display the hypermedia, the processor being operable to identify elements that define predetermined keyboard shortcuts in the hypermedia, and form an options list containing data associated with the identified elements, and the processor and the display device being operable in a second display configuration to display the options list.

## BRIEF DESCRIPTION OF THE DRAWINGS

In order that the invention may be more fully understood, an embodiment will now be described with reference to the accompanying drawings in which; Figure 1 is a schematic block diagram illustrating a mobile telephone handset that can communicate through a PLMN to a server;

Figure 2 is a schematic block diagram of a the circuitry of the mobile handset shown in Figure 1;

Figure 3 illustrates schematically that the display of the handset and the associated keys 4;

Figure 4 illustrates a scrolling functionality associated with key 4c;

Figure 5 is a schematic illustration of the process performed by the browser;

Figure 6A and 6B illustrate screen displays for the mobile telephone handset when using either a prior WML browser, or a XHTML browser in accordance with the invention;

Figure 7 is a schematic flow diagram of the process steps carried out when parsing an XHTML document in accordance with the invention;

Figure 8 is a screen display for a second example of the invention; Figure 9 is a schematic illustration of an XHTML document; and

Figure 10 is a schematic illustration of the options list provided in accordance with the invention for the XHTML document shown in Figure 9.

5

#### DETAILED DESCRIPTION

In Figure 1, a mobile station in the form of a battery driven telephone cellular handset MS 1, is shown schematically in radio communication with PLMN 1. A GSM network is described herein by way of example but it will be understood that other  
10 PLMNs could be used, utilizing CDMA, TDMA or other transmission techniques. The mobile handset MS1, includes a microphone 2, keypad 3, soft keys 4, a liquid crystal display device 5, ear-piece 6 and internal antenna 7. As will be explained in more detail, the handset 1 is enabled with a user agent in the form of a browser for browsing hypermedia.

15 The circuitry of the handset MS 1 is shown in more detail in Figure 2. Signal processing is carried out under the control of a digital micro-controller 8 that has an associated RAM/ROM 9 and flash memory 10. Electrical analogue audio signals are produced by microphone 2 and amplified by pre-amplifier 11. Similarly, analogue audio signals are fed to the earpiece 6 through an amplifier 12. The micro-controller receives  
20 instruction signals from the keypad and so-called soft keys 4 and controls operation of the LCD display 5.

Information concerning the identity of the user is held on a smart card 13 in the

form of a GSM SIM card, which contains the usual GSM international mobile subscriber identity and encryption  $K_1$  that is used for encoding the radio transmission in a manner well known *per se*. The SIM card 13 is removably received in a SIM cardholder 14.

Radio signals are transmitted and received by means of the antenna 7 connected  
 5 through a rf stage 15 to a codec 16 configured to process signals under the control of a micro-controller 8. Thus, in use, for speech, the codec 16 receives analogue signals from the microphone amplifier 11, digitizes them into a form suitable for transmission and feeds them to the rf stage 15 for transmission through antenna element 7 to PLMN 1 shown in Figure 1. Similarly, signals received from PLMN I are fed through the  
 10 antenna element to be demodulated by the rf stage 15 and fed to codec 16 so as to produce analogue signals fed to amplifier 12 and earpiece 6.

The handset 1 is enabled to receive and display hypermedia in a browser. Data can be downloaded to the browser from a server 18 shown in Figure 1. The server 18 provides hypermedia in the form of pages that can be downloaded to the handset MSI  
 15 and manipulated for display by means of the keys 3, 4. The server 18 may be accessed through a gateway (not shown), which acts as a proxy server. The handset MS I can access the gateway by dialing a predetermined telephone number.

The hypermedia downloaded from the server 18 may be held in the RAM 9 or the flash memory 10. The microcontroller 8 provides the browser functionality and causes  
 20 pages to be presented to the user in the browser on the screen of the display 5, and the display can be manipulated by means of the keys 3, 4.

The functionality of the soft keys 4 will now be described in more detail with

reference to Figure 3. The keys 4 are disposed adjacent the underside of the display 5. The keys 4a, 4b comprise manually depressible buttons. Their functionality can be pre-programmed depending upon the task performed. The function attributed to the keys is displayed on the display 5 as soft key function legends 20, 21 adjacent to the respective  
5 keys 4a, 4b.

The key 4c comprises a roller device to perform a scrolling function for the display 5. In use, an active or focus region 22 is provided on the display which, as explained in more detail later, can be moved around the display area. The roller 4c may comprise a 3-way roller, as shown in Figure 4. The roller may be pushed by the user's  
10 finger, to pivot about axis 23. When pushed upwardly, in the direction of arrow 24, a "scroll-up" function is performed. Conversely, when pushed downwardly in the direction of arrow 25 a "scroll-down" function is performed. Additionally, the roller 4c may be manually depressible inwardly of the handset, in the direction of arrow 26, to perform a "select" function. The key 4c may be further configured as a five-way roller, not shown,  
15 so as additionally to perform right and left scrolling functions. Alternatively, the key 4c may comprise a rollerball, a touch pad or other navigation device of the type used in laptop computers. In this example, the focus region 22 is delineated by a rectangular box which is moved around the screen of the display 5. However, the focus region can be provided in different ways, such as a region highlighted with a different intensity or  
20 colour from the rest of the display, an underlining of a menu option or by means of a pointer movable across the display in the manner of a conventional mouse pointer.

It is known in the art that the handset I may be WAP enabled and capable of



receiving data in a predetermined channel. For example, in GSM, a data channel is available at 9.6 kbit/sec. Also, the handset may be configured to receive high speed circuits switch data (HSCSD) according to the GSM recommendations. It would also be possible to use GPRS and formats appropriate to 3G, or other data transmission

5 techniques such as i-mode. The WAP content and its applications are specified in a well known set of content formats based upon familiar www content formats. WAP is disclosed in the wireless application protocol architecture Specification; Version 1.3, published by The Wireless Application Protocol Architecture Working Group, 19 February 2000.

10 The browser operated by the handset 1, may comprise a micro-browser that acts as a client in the WAP environment for connection to WAP servers.

## WML

Conventionally, the browser is configured for use with WML, which comprises a  
15 lightweight mark-up language similar to HTML but optimiszed for handheld mobile terminals. This will now be described in more detail to explain the problem solved by the invention.

In order to accommodate the relatively small display areas provided in mobile devices, a conventional page of HTML is broken down into smaller units known as  
20 cards, arranged in a deck.

The deck commences with a prologue and is delimited by a <wm1> tag. An example of the syntax of a deck of two cards is set out below.

**WML code snippet No.1**

```
<wml>
<template>
5   <do type="prev" label="Back"><prev/></do>
</template>
<card id="card1" title="Card 1">
  <do type="accept" label="Card 2">
    <go href="#card2"/>
10  </do>
    <p>
      Hello World!
      This is the first card.....
    <p>
15 </card>
<card id="card2" title="Card 2">
  <p>
    This is the second card.
    Goodbye.
20 </p>
  </card>
</wml>
```

Elements and attributes of WML code will now be explained with reference to the example.

### **template element**

5       The <template> element contains optional information about deck-level bindings.

### **card element**

      The <card> element defines the user interface and processing logic of the deck. As previously mentioned, the deck may include a number of cards and in this example  
10   two cards are included.

### **DO element**

      The DO element provides a general mechanism for the user to act upon the current card, and is mapped onto a particular user agent widget, which is determined by the specific user agent i.e. the browser. In the Nokia WAP browser, the DO elements  
15   are mapped onto a list in an options menu, which is displayed by operating the right softkey 4a (except for the element <do type="prev" label="Back"><prev/></do>, which is displayed as the softkey function legend 21 for the left softkey 4b, to provide a "Back" functionality associated with the key 4b).

      In this example, a DO element triggers a GO element that goes to the universal  
20   resource locator (url) of the second card of the deck, and causes the second card to be displayed.

      The DO element can be included at the card level or can be included in the

template to operate on all cards in the deck.

### **ANCHOR element**

As with HTML, links can be defined. In WML, the links can be between decks or  
5 cards. The links are defined in WML by an ANCHOR element that defines the head of  
the link. WML also includes an element A, which is a shorthand version of the ANCHOR  
element. This is not shown in WML code snippet No.1 above, but will be explained in  
more detail later.

### 10 **INPUT element**

Furthermore, WML allows a user to insert text, for example to enter a user name,  
by means of an element INPUT. This element thus specifies a text entry object, in a  
similar manner to HTML.

### 15 **accesskey attribute**

As in HTML, certain WML elements have an attribute that assigns an access key  
to the element. In this way, the WML elements can be given a particular focus by  
operating a predetermined key on the keyboard 3,4 of the device. The attribute is  
defined in WML as accesskey and is an attribute of the WML elements: ANCHOR, A  
20 and INPUT.

An example is given below of the use of the accesskey element in a link using  
the A element (with A being shown in lower case i.e. "a" below):

<a accesskey="1"

href=http://someplace.com/specification/contents.html> Table of Contents</a>

5           The accesskey functionality is invoked by actuating the key concerned on the keypad. Thus, in this example, when the key "1" on the keypad is pressed, when the link "someplace" is within the focus region 22, the accesskey functionality is invoked and a link to the domain "someplace" is actuated, to locate the contents of a table.

          A more detailed description of these WML elements and others is given in  
10 "Official Wireless Application Protocol" Wiley, 1999 [ISBN 0-471-32755-7].

          As previously mentioned, the small size of the display 5 used in portable devices means that only part of the deck can be displayed at any time so that links contained within the deck are not necessarily visible whilst the user browses the content of the deck. Hitherto, proprietary WML browsers have been configured so that one of the soft  
15 keys 4a, 4b is configured to offer the user an options list, in which, links associated with a no element are listed in a separate options screen so that all relevant links can be reviewed by the user without having to browse through the entire deck of cards. WML code snippet number 2 given below illustrates an example of how the DO element can be used in this way.

20

### **WML code snippet No.2**

<wml>

```

5  <card id="Cnnnews" title="CNN News">
    <p>
        <a href="domnews.wml">Domestic news</a>
        <a href="fornews.wml">Foreign news</a>
        <a href="sports.wml">Sports</a>
        <a href="markets.wml">Markets</a>
    </p>
    <do type="accept" name="home" label="CNN Home">
        <go href="Cnnmain.wml"/>
10 </do>
    <do type="accept" name="help" label="CNN Help">
        <go href="helpbusi.wml"/>
    </do>
</card>
15 </wm 1>

```

The process performed by the browser is shown schematically in Figure 5 and comprises a main browser process 27, which parses the WML code, soft key function process 28 which responds to actuation of the soft keys 4a - c and an options process 29. The outcome of the processing is rendered to the user by a display process 30 in order to provide a display on the display screen 5.

These processes render the code of WML code snippet No. 1 to provide a display as shown in Figure 6A on the display screen 5. The soft key function process 28

designates keys 4a and 4b as “options” and “back” 20a, 21a. When the user operates the “options” key 4a, the display shown in Figure 6B is provided on the display screen 5, because options process 29 was selected. The options process 29 identifies events in the code snippet 2 that are delineated by the DO element. The DO elements so

5 identified are then rendered in the display of Figure 6B by the options process 29. The display of Figure 6B may also include other options that are to be displayed, produced by other means. Thus, in Figure 6B, the anchors “CNN Home” and “CNN Help” are displayed, because they are associated with DO elements in WML code snippet 2. The display of Figure 6B also includes “bookmarks” and “use number” which are generated

10 separately by means of the browser process 27.

The provision of the DO element items in the options list of Figure 6B has the advantage that individual links can be provided in the options list and so that the user can identify them without having to browse the entire deck of cards. Thus, by selecting the options display of Figure 6B, the user can review the links irrespective of the

15 position in the current card.

An example of use of the WML access key will now be described.

### **WML code snippet No.3**

```

<wml>
20  <card id="Cnnnews" title="CNN News">
    <p>
        <a accesskey="l" href="domnews.wml">Domestic newsc/a>

```

```
<a accesskey="2" href="fornews.wml">Foreign news</a>
```

```
<a accesskey="3" href="sports.wml">Sports</a>
```

```
<a accesskey="4" href="markets.wml">Markets</a>
```

```
</p>
```

```
5 <do type="accept" name="home" label="CNN Home">
```

```
<go href="Cnnmain.wml"/>
```

```
</do>
```

```
<do type="accept" name="help" label="CNN Help">
```

```
<go href="helpbusi .wml"/>
```

```
10 </do>
```

```
<card>
```

```
</wml>
```

As previously explained, the accesskey attribute allows individual keys of the keypad to be assigned to particular links, to provide a shortcut. Thus, in the described example, the keys 1, 2, 3, and 4 of keypad 3 are assigned individual accesskey functions. The code snippet 3 is processed by the browser in a similar manner to that shown in Figure 4 and the rendered displays are the same as shown in Figure 6A and 6B.

## 20 XHTML

It has recently been proposed to develop XHTML browsers for use with mobile devices such as mobile telephone handsets and PDAs. XHTML is an extension of



HTML 4, which in turn is an SGML (standard generalized markup language) conforming to international standard ISO 8879 and is widely regarded as a standard publishing language for the World Wide Web. XHTML has been designed with alternate ways of accessing the Internet in mind and the XHTML family is designed with user agent interoperability in mind. However, a disadvantage of XHTML is that it does not include a DO element with functionality as described hereinbefore in relation to WML. This has the disadvantage that if a long text document is provided in XHTML, the user needs to browse through the entire document to determine the available links. There is no display that corresponds to the display of Figure 6B, described with reference to WML. The invention seeks to overcome this problem.

XHTML does however include the accesskey attribute. The accesskey functionality may be invoked as previously described.

According to the invention, the accesskey attribute is parsed in such a way as to provide a separate list of links available in the document. An example of XHTML code for use according to the invention is set out below as XHTML code snippet No 1.

#### **XHTML code snippet No.1**

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>CNN News</title>
  </head>
  <body>
```

<p>

1. <a href="domnews.html" accesskey="1">Domestic  
news</a><br/>

2. <a href="fornews.html" accesskey="2">Foreign  
5 news</a><br/>

3. <a href="sports.html" accesskey="3">Sports</a><br/>

4. <a href="markets.html" accesskey="4">Markets</a><br/>

<a href="cnmain.html" accesskey="5" style="visibility:  
hidden">CNN Home</a>

10 <a href="helpbusi.html" accesskey="6" style="visibility:  
hidden">CNN Help</a>

</p>

</body>

</html>

15 According to the invention, the browser process 27 of Figure 5 is configured to  
parse XHTML hypermedia and the display process 30 renders a display corresponding  
to Figure 6A and 6B. According to the invention, links that are marked with accesskey  
attributes are provided in the options display of Figure 6B. Thus, the previously  
described WML functionality is provided in XHTML but with out the use of a DO  
20 element.

In the case where style="visibility:hidden" is defined, the actual link will not be  
visible in the display, but will still be included in the options list. This is identical to WML

DO element behaviour, which does not have any visible rendering within the document itself. The other accesskey cases in the above code snippet, 'without "visibility:hidden" show how the invention is simpler and more powerful than WML DO, since it combines both the <a> (anchor) functionality and DO functionality.

5       When style="visibility:hidden" is set for a particular accesskey so that it is not visible in Fig. 6A, its associated link can however be selected by performing a relatively long key press of the associated key on the keyboard. For example, in XHTML code snippet No.1, the link "CNN Home" associated with accesskey = "5" is hidden in the display of Fig. 6 but can be selected by a long press of the 5 key of the keyboard.

10       The process performed by the browser will now be described in more detail with reference to Figure 7. At step S1, the XHTML document is parsed and at step S2, items that have been marked with an accesskey element are identified. The links associated with the elements naming an accesskey attribute are then included in an options list at step S3. As previously explained, the option process 29 shown in Figure 5 may produce  
15 an options list that includes not only the access key elements but also other links which are provided by other processes, such as "bookmarks" and "use number".

When the user operates the options key, namely soft key 4a with functionality 20a shown in Figure 6A, the options list of Figure 6B is displayed as shown at step S5.

Individual links from the display of Figure 6B can then be selected by scrolling  
20 the focus region 22 up or down with key 4c, in the direction of arrows 24, 25 (Figure 4) in order to select a particular link. The link may then be actuated by depression of the key 4c in the direction of arrow 26. Alternatively, another key on the handset may be

used for actuation of the link, such as the usual “send” or “place call” key (not shown).

Referring in more detail to XHTML code snippet No. 1, it will be seen that the accesskey functions 1 and 2 have been assigned to the link corresponding to “CNN Home” and “CNN Help” respectively and hence these two links are displayed in the options list of Figure 6B. It will be seen that the style has been set to “visibility: hidden” with a result that link associated with the accesskey functionality is suppressed in the display of Figure 6A. The “visibility: hidden” attribute is a feature of XHTML and defines the particular template that is to be used for display of the element.

Thus, the user can access these links independently of their location in the XHTML page and independently of the position of the page in the XHTML browser.

This advantage will be seen with greater clarity with reference to the following XHTML code snippet and its corresponding display.

#### **XHTML code snippet No.2**

```

15 <html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Specimen page of textual information
            containing linksc/title>
    </head>
20 <body>
    <p>
        <a href="morenews.html" accesskey="l">more news</a>

```

</p>

<p> This is an example of a long item of text which fills up all of the display screen of a mobile telephone handset and which cannot all be displayed at once. The item of text contains so many words that it would not be possible to fit it all within the limited display  
5 screen size usually found on a mobile telephone handset. </p>

<p> Some items such as news items, require a significant number of words to describe the details of the story that is being conveyed to the reader. </p>

<p>The item may include links to other pages, such as a link to a page with information relating to “sports”, the link being provided below.</p>

10 <p>

<a href="sports.html" accesskey="2">Sports</a>

<a href="anotherlink.html">Another link</a> </p>

<p>The link may be located in the text at a particular point dependent upon the context, or at the beginning or the end. </p>

15 <p>The item may also include graphics and other items such as tables. </p>

<p>Some more text is hereby provided after which some more links are shown. When browsing the page, the user cannot tell whether there are links in the text or whether there are more links at the end, unless the entire page is viewed. . </p>

<p>

20 <a href="homepage.html" accesskey="3">homepage</a>

<a href="markets.html" accesskey="4">markets</a>

</p>

</body>

</html>

The display corresponding to the code is shown in Figures 8, 9 and 10. The XHTML code is parsed and processed as shown in Figure 7 and the initial display provided on display screen 5 is shown in Figure 8. Figure 9 illustrates schematically that the entire HTML document cannot be displayed in the relatively small area of the screen 5 but that the document can be scrolled using the soft key 4c so that all of the XHTML document can be made available for display.

It will be seen that in the display of Figure 8, only one of the available links is initially visible to the user, namely "More News". However, from the display of Figure 9, it will be seen that there are four further links that are available to the user when the document is scrolled. The links "Sports" "Homepage" and "Markets" are each provided with an individual accesskey code. As a result, when the options key 4a, 20a is actuated (Step S4 of Figure 7) the links that have been marked with accesskey elements are displayed in a separate options list as shown in Figure 2. Thus, the user is provided with a convenient list of all the links with associated access keys irrespective of the position of the document in the browser, thus overcoming the need to scroll through the entire document to identify all of the links.

It will be noted that not all of the links need be included in the options list of Figure 10. The "another link" shown in Figure 9 is not marked with an accesskey function in the XHTML code and as a result is not included in the display of Figure 10. This is under the control of the programmer.

Many modifications and variations of the invention falling 'within the spirit and scope of the claims are possible. For example, because the accesskey functionality is also available in WML and SGML, the invention is not restricted to XHTML but can be used with other mark up languages.

**ABSTRACT**

A browser parses an XHTML document to identify links in the document that are marked up with the accesskey attribute and forms an options list containing the links, so that they can be selected independently of their position in the document.